What is claimed is:

1. An apparatus, comprising:

a first circuitry to permit access, responsive to entering a first password, to a first set of resources to debug a first set of code in a memory;

a storage structure to contain a second password; and

a second circuitry coupled to the first circuitry and to the storage structure to permit access, responsive to entering the second password, to a second set of resources to debug a second set of code in the memory.

2. The apparatus of claim 1, wherein said access to the second set of resources is through a debug interface.

3. The apparatus of claim 2, adapted to place the second set of code in a substantially different portion of the memory than the first set of code.

4. The apparatus of claim 1, wherein the storage structure comprises a content addressable memory.

5. The apparatus of claim 1, further comprising circuitry to prevent the access responsive to said entering the first password if the access responsive to said entering the second password is enabled, and to prevent the access responsive to said entering the second password if the access responsive to said entering the first password is enabled .

6. The apparatus of claim 1, wherein the storage structure comprises a programmable storage structure.

7. The apparatus of claim 1, wherein said first set of resources comprises a first portion of the memory and said second set of resources comprises a second portion of the memory different than the first portion.

8. The apparatus of claim 1, wherein the second set of resources is substantially a subset of the first set of resources.

9. A system, comprising:

a volatile first memory;

a second memory coupled to the first memory to contain code for execution;

a processor coupled to the second memory to execute the code;

a first storage structure coupled to the processor to contain a first password;

a second storage structure coupled to the processor to contain a second password; and

circuitry to permit access, responsive to entering the first password, to a first set of resources to debug a first set of code in the second memory, to disable said access to the first set of resources, and to enable access, responsive to entering the second password, to a second set of resources to debug a second set of code in the second memory.

10. The system of claim 9, wherein the first and second access are to be through a debug interface.

11. The system of claim 9, wherein the second storage structure is a content addressable memory.

12. The system of claim 9, wherein the circuitry is adapted to cause the access responsive to said entering the first password and the access responsive to said entering the second password to be mutually exclusive.

13. A method, comprising:

disabling a first password that enables performing a first set of code debug operations;

storing a second password; and

entering the second password to enable performing a second set of code debug operations;

wherein being enabled to perform the first set of code debug operations and being enabled to perform the second set of code debug operations are mutually exclusive.

14. The method of claim 13, wherein said being enabled to perform the second set of code debug operations comprises being enabled to perform a subset of the first code debug operations.

15. The method of claim 13, wherein said first and second code debug operations are performed through a debug interface.

16. The method of claim 13, wherein said disabling the first password results from said storing the second password.

17. The method of claim 13, further comprising enabling a third password that re-enables said performing the first set of code debug operations.

18. A machine-readable medium that provides instructions, which when executed by a computing platform, cause said computing platform to perform operations comprising:

receiving a first password to enable debug of a first set of code during a first debug stage;

disabling the first password to prevent further debugging activities during the first debug stage;

storing a second password; and

receiving the second password to enable debug of a second set of code during a second debug stage.

19. The medium of claim 18, wherein the operation of disabling the first password prevents access to the first set of code during the second debug stage.

20. The medium of claim 18, wherein the operation of storing the second password results in said disabling the first password.

21. The medium of claim 18, wherein said operations further comprise using a third password to re-enable the debug of the first set of code.

22. An apparatus, comprising:

a processor to execute instructions in a debug mode;

circuitry to contain control bits; and

circuitry coupled to the processor and to the first circuitry to determine a level of debug access permitted in the apparatus based on a state of the control bits.

23. The apparatus of claim 22, wherein said circuitry to contain control bits comprises a set of programmable fuses.

24. The apparatus of claim 22, further comprising an interface coupled to the processor to provide said debug access.

25. The apparatus of claim 22, wherein said circuitry to determine a level of debug access is to determine restrictions for at least one of:

an address range within which said at least one instruction may be located;

which instructions may be executed; and

resources which may be accessed resultant to said execution.

26. The apparatus of claim 25, further comprising circuitry to abort the debug access resultant to said debug access violating at least one of said restrictions.

27. A system, comprising:

a processor to execute instructions in a debug mode;

a volatile memory coupled to the processor;

circuitry to contain a set of control bits to indicate a level of debug access permitted during the debug mode based on a state of at least one of the control bits; and

circuitry to determine if the execution violates restrictions defined by the level of debug access.

28. The system of claim 27, wherein the state of the control bits is non-alterable after manufacture of the circuitry.

29. The system of claim 27, further comprising circuitry to present a result of the execution to a debug interface if the execution does not violate the restrictions and to not present the result of the execution to the debug interface if the execution violates the restrictions.

30. The system of claim 29, wherein the circuitry to present the result comprises circuitry to abort the debug mode if the execution violates the restrictions.

31. The system of claim 27, wherein the circuitry to determine if the execution violates the restrictions is adapted to make said determination after execution of a predetermined group of at least one instruction.

32. A method, comprising:

determining, based on a set of control bits, a current level of authorized debug access;

executing at least one instruction in a trusted subsystem in a debug mode of operation; and

not presenting a result of said executing if said executing violates restrictions defined by the current level of authorized debug access.

33. The method of claim 32, wherein said not presenting comprises aborting the debug mode of operation.

34. The method of claim 32, further comprising presenting the result of said executing if said executing does not violate the restrictions defined by the current level of debug access.

35. The method of claim 32, wherein the restrictions are based on at least one of:

an address of the at least one instruction being within a particular memory address range;

the at least one instruction being within a predetermined list of instructions; and

all resources accessed by said executing being within a predetermined list of resources.